# Nmap Basics, Target Specification and Port States

**1.** The command "nmap -version" is used to check the version of Nmap installed on your system.

When you run the command "nmap -version" in the command line or terminal, Nmap will display the version information, including the version number and other details about the installed Nmap software.

Here is an example of how to use the command:

```
nmap –version
```

After executing the command, Nmap will display the version information, such as:

```
Nmap version 7.91 ( https://nmap.org )
Platform: x86_64-pc-linux-gnu
Compiled with: liblua-5.3.3 openssl-1.1.1g libssh2-1.8.0 libz-1.2.11 libpcre-8.44
libpcap-1.10.0 nmap-libdnet-1.12 ipv6
Compiled without:
Available nsock engines: epoll poll select
```

The displayed information will vary depending on the version of Nmap installed and the platform it is running on.

---

**2.** The command "nmap -h" is used to display the help menu and provide information about the usage and options of the Nmap command-line tool.

```
nmap –h
```

When you run the command "nmap -h" in the command line or terminal, Nmap will display a list of available command-line options and their descriptions.

Here is an example of how to use the command:

---

**3.** To scan a single IP address using Nmap, you can use the following command:

```
nmap <IP_ADDRESS>
```

Replace `` with the actual IP address you want to scan.

For example, if you want to scan the IP address 192.168.1.35, you would run the command:

```
nmap 192.168.1.35
```

---

**4.** To scan multiple IP addresses using Nmap, you can provide a list of IP addresses separated by spaces in the command. Here's an example:

```
nmap <IP_ADDRESS_1> <IP_ADDRESS_2> <IP_ADDRESS_3> ...
```

Replace `, `, `, and so on, with the actual IP addresses you want to scan.

For instance, if you want to scan three IP addresses (192.168.1.35, 192.168.1.36, and 192.168.1.37), you would run the command:

```
nmap 192.168.1.35 192.168.1.36 192.168.1.37
```

When you execute this command, Nmap will perform scans on each of the specified IP addresses and provide information about open ports and services running on each target system. The output will show the results of the scan for each IP address individually.

---

**5.** To scan a range of IP addresses using Nmap, you can specify the range using the IP address format. Here's an example:

```
nmap <START_IP_ADDRESS>-<END_IP_ADDRESS>
```

Replace ` with the starting IP address of the range, and ` with the ending IP address of the range.

For example, if you want to scan a range of IP addresses from 192.168.1.1 to 192.168.1.10, you would run the command:

```
nmap 192.168.1.1-10
```

---

**6.** To perform a verbose scan using Nmap, you can use the "-v" option. This option provides more detailed output during the scan. Here's an example:

```
nmap -v <IP_ADDRESS>
```

Replace ` with the actual IP address you want to scan.

For instance, if you want to perform a verbose scan on the IP address 192.168.1.35, you would run the command:

```
nmap -v 192.168.1.35
```

When you execute this command, Nmap will initiate the scan and provide more detailed information about the progress of the scan, open ports, services, and other relevant details. The verbose output will help you understand the scan process and the results more thoroughly.

---

**7.** To scan a domain using Nmap, you can provide the domain name instead of an IP address. Nmap will resolve the domain name to its corresponding IP address and perform the scan on that IP address. Here's an example:

```
nmap <DOMAIN_NAME>
```

Replace `` with the actual domain name you want to scan.

For example, if you want to scan the domain "example.com", you would run the command:

```
nmap example.com
```

---

**8.** To scan a network range in CIDR notation using Nmap, you can specify the CIDR range as the target in the command. Here's an example:

```
nmap <CIDR_RANGE>
```

Replace `` with the actual CIDR range you want to scan.

For instance, if you want to scan the network range 192.168.0.0/24, you would run the command:

```
nmap 192.168.0.0/24
```

When you execute this command, Nmap will perform scans on each IP address within the specified CIDR range and provide information about open ports and services running on each target system. The output will show the results of the scan for each IP address individually.

---

**9.** To scan targets from a file using Nmap, you can provide the file path containing the list of targets using the "-iL" option. Here's an example:

```
nmap -iL <FILE_PATH>
```

Replace `` with the actual path to the file containing the list of targets.

For example, if you have a file named "targets.txt" located in the current directory which contains a list of IP addresses or domain names, you would run the command:

```
nmap -iL targets.txt
```

When you execute this command, Nmap will read the list of targets from the file and perform scans on each target. It will provide information about open ports and services running on each target system. The output will show the results of the scan for each target individually.

---

**10.** To scan random hosts using Nmap, you can use the "--randomize-hosts" option. This option instructs Nmap to randomize the order in which it scans the hosts. Here's an example:

```
nmap --randomize-hosts -iR <NUMBER_OF_HOSTS>
```

Replace `` with the desired number of random hosts you want to scan.

For instance, if you want to scan 10 random hosts, you would run the command:

```
nmap --randomize-hosts -iR 10
```

When you execute this command, Nmap will randomly select and scan the specified number of hosts. It will provide information about open ports and services running on each randomly selected host. The output will show the results of the scan for each host individually.

---

**11.** To exclude specific hosts from an Nmap scan, you can use the "--exclude" option followed by a comma-separated list of hosts or IP addresses you want to exclude. Here's an example:

```
nmap --exclude host1,192.168.0.100,10.0.0.1-10 target
```

In the above example, the Nmap scan is performed on the target network or host, excluding "host1", "192.168.0.100", and the IP range from "10.0.0.1" to "10.0.0.10". Replace "target" with the actual IP address or hostname you want to scan.

By using the "--exclude" option, Nmap will skip scanning the excluded hosts and only focus on the remaining hosts or IP addresses. This can be useful when you want to exclude specific systems from a scan, such as trusted or known hosts that don't require scanning.

---

**12.** In Nmap, the -sS option is used to perform a "TCP SYN Scan." When you use the -sS option, Nmap sends TCP SYN packets to the target ports and analyzes the responses received. This type of scan is also known as a "half-open scan" or "stealth scan."

The TCP SYN scan is one of the most popular and widely used scan techniques in Nmap. It works by sending a SYN packet to the target port and analyzing the response received. If the port is open, the target responds with a SYN/ACK packet. If the port is closed, the target responds with a RST (reset) packet. By analyzing these responses, Nmap can determine if a port is **open, closed, or filtered by a firewall.**

```
nmap -Ss <target>
```

Replace `` with the IP address or hostname of the target you want to scan. Nmap will perform a TCP SYN scan on the target ports and provide a report of the open, closed, and filtered ports.

It's important to note that the TCP SYN scan is considered stealthy because it does not complete the full TCP handshake, making it less likely to be detected by intrusion detection systems (IDS) or firewalls. However, some systems and networks may still log or block SYN packets, so it's important to use Nmap responsibly and with proper authorization.

---

**Question #1:**

What is the firewall?

**Answer #1:**

A firewall is a security device or software that monitors and controls network traffic between different networks, typically an internal network and an external network like the internet. It acts as a barrier, allowing or blocking network traffic based on predefined rules to protect the internal network from unauthorized access and potential threats. Firewalls can be implemented at different network layers and help to secure networks by filtering and inspecting incoming and outgoing packets, enforcing security policies, and preventing malicious activities.